

Tight Hardness Results for Distance and Centrality Problems in Constant Degree Graphs

Søren Dahlgaard^{*1} and Jacob Evald¹

¹University of Copenhagen , [soerend, jeav]@di.ku.dk

September 28, 2016

Abstract

Finding important nodes in a graph and measuring their importance is a fundamental problem in the analysis of social networks, transportation networks, biological systems, etc. Among the most popular such metrics of importance are graph centrality, betweenness centrality (BC), and reach centrality (RC). These measures are also very related to classic notions like diameter and radius. Roditty and Vassilevska Williams [STOC'13] showed that no algorithm can compute a $(3/2 - \delta)$ -approximation of the diameter in sparse and unweighted graphs faster than $n^{2-o(1)}$ time unless the widely believed strong exponential time hypothesis (SETH) is false. Abboud et al. [SODA'15] and [SODA'16] further analyzed these problems under the recent and very active line of research on hardness in **P**. They showed that in sparse and unweighted graphs (weighted for BC) none of these problems can be solved faster than $n^{2-o(1)}$ unless some popular conjecture is false. Furthermore they ruled out a $(2 - \delta)$ -approximation for RC, a $(3/2 - \delta)$ -approximation for Radius and a $(5/3 - \delta)$ -approximation for computing all eccentricities of a graph for any $\delta > 0$.

In this paper we extend these results to the case of unweighted graphs with constant maximum degree. Through new graph constructions we are able to obtain the same approximation and time bounds as for sparse graphs even in unweighted bounded-degree graphs. Specifically we show that no $(3/2 - \delta)$ approximation of Radius or Diameter, $(2 - \delta)$ -approximation of RC, $(5/3 - \delta)$ -approximation of all eccentricities or exact algorithm for BC exists in time $n^{2-o(1)}$ for such graphs and any $\delta > 0$. This strengthens the result for BC of Abboud et al. [SODA'16] by showing a hardness result for unweighted graphs, and follows in the footsteps of Abboud et al. [SODA'16] and Abboud and Dahlgaard [FOCS'16] in showing conditional lower bounds for restricted but realistic graph classes.

1 Introduction

Measuring the importance of specific nodes in a graph is a fundamental problem in the analysis of social networks, transportation networks, biological systems, etc. Several notions of importance have been proposed in the literature. Among the most popular such notions are several centrality measures such as *graph centrality* [19], *betweenness centrality* (BC) [16], *closeness centrality* [27], and *reach centrality* [18]. All these centrality measures are closely related to the shortest paths of the graph. As an example, the graph centrality of a node is the inverse of its maximum distance to another node in the graph. These measures are extensively studied in both the theoretical and practical communities with some papers having thousands of citations [10, 16, 20, 24]. Fully

^{*}Research partly supported by Mikkel Thorup's advanced grant DFF-0602-02499B from the Danish Council for Independent Research.

understanding the complexity of computing these measures is thus a very important problem in the field of graph analysis. In this paper we follow a recent and very active line of research on showing hardness or *conditional lower bounds* (CLBs) for problems in \mathbf{P} [25, 4, 6, 3, 14, 5, 2, 26] focusing on the above mentioned centrality measures in graphs with constant degrees.

Given an undirected and unweighted graph $G = (V, E)$ with n nodes and m edges we let $d_G(u, v)$ denote the distance between nodes $u, v \in V$. We omit the subscript G , when it is clear from the context. Let $\sigma_{s,t}(u)$ denote the number of distinct shortest paths between s and t and let $\sigma_{s,t}(u)$ denote the number of such paths passing through u . In this paper we consider several centrality and importance measures of graphs. We summarize the definitions of these below in Table 1. We note that all of these definitions also make sense for weighted graphs, but we concentrate on unweighted graphs in this paper, as they are more difficult to show hardness results for in general (see eg. [2]). We note that the maximum Graph Centrality is exactly the

Name	Definition
The <i>Eccentricity</i> (of u)	$\epsilon(u) := \max_{v \in V} d(u, v)$
The <i>Diameter</i>	$D := \max_{u \in V} \epsilon(u)$
The <i>Radius</i>	$R := \min_{u \in V} \epsilon(u)$
The <i>Reach Centrality</i> (RC) (of u)	$RC(u) := \max_{\substack{s, t \in V \\ d(s, t) = d(s, u) + d(u, t)}} (\min(d(s, u), d(u, t)))$
The <i>Betweenness Centrality</i> (BC) (of u)	$BC(u) := \sum_{s, t \in V \setminus \{u\}} \frac{\sigma_{s,t}(u)}{\sigma_{s,t}}$
The <i>Graph Centrality</i> (GC) (of u)	$GC(u) := \frac{1}{\max_{v \in V} d(u, v)}$

Table 1: Definitions of different centrality and importance measures.

inverse of the Radius. All of these measures except BC can be computed by simply running an algorithm for the classical all pairs shortest paths (APSP) problem in $\tilde{O}(n^\omega)$ or $\tilde{O}(mn)$ time, where ω is the matrix-multiplication exponent. For BC we can use Brandes's algorithm [10] to compute the betweenness centrality of all nodes in $O(mn + n^2 \log n)$ time¹.

1.1 Hardness in \mathbf{P}

A recent and very active line of work concerns itself with showing hardness results for problems in \mathbf{P} based on the assumption of several popular conjectures. For the measures in Table 1 several results are known. Perhaps the most well-studied of the problems from a theoretical perspective is diameter. Roditty and Vassilevska Williams [26] showed that no algorithm can solve the diameter problem in sparse graphs in time $O(n^{2-\epsilon})$ for any $\epsilon > 0$ unless the widely believed Strong Exponential Time Hypothesis (SETH) [23] is false. In fact they showed that no algorithm can even compute a $3/2 - \delta$ approximation in this time for any $\delta > 0$. We say that a number x is an α -approximation of a number y if $y \leq x \leq \alpha y$. In some cases we will also allow the algorithm to provide an under-approximation. Abboud et al. [5] showed similar results for the problem of computing the radius, median, and all eccentricities in sparse graphs. They showed that unless SETH is false no algorithm can compute a $5/3 - \delta$ approximation of all eccentricities in time $O(n^{2-\epsilon})$ for any $\epsilon, \delta > 0$ in sparse graphs. Based on the similar *Hitting Set (HS) Conjecture* they also showed hardness results for the radius and median problems. For radius, they showed that no algorithm can compute a $3/2 - \delta$ approximation in time $O(n^{2-\epsilon})$ for any $\epsilon, \delta > 0$.

For the centrality measures of Table 1, Abboud et al. [3] showed that radius, median, and

¹We note that Brandes's and other popular algorithms for computing BC neglects the complexity of keeping the counters of the number of shortest paths. If taking this into account the worst-case running time grows by a factor of $\Theta(n \log n)$.

betweenness centrality are all equivalent to the classic APSP problem under *subcubic reductions*. Similarly, they showed that RC is equivalent to the diameter problem under subcubic reductions. For betweenness centrality they showed that computing an α -approximation for any $\alpha > 0$ is equivalent to APSP under subcubic reductions and that no algorithm can compute such an approximation for any node in time $O(n^{2-\varepsilon})$ in sparse graphs unless SETH is false. Finally they show that computing reach centrality is equivalent to diameter under subcubic reductions and that computing a $(2 - \delta)$ -approximation of RC in sparse and *unweighted* graphs cannot be done in $O(n^{2-\varepsilon})$ time unless SETH is false. An important note about all these reductions except for RC in sparse graphs is that they only hold for *weighted* graphs. They mention in [3] that it is an interesting open problem to show a CLB for approximating BC in unweighted graphs.

The known hardness results for the measures in Table 1 are summarized below in Table 2.

Problem	Bound	Approximation	Graph family	Source
Diameter	$n^{2-o(1)}$	$3/2 - \delta$ -approx	Sparse, unweighted	[26]
Radius	$n^{2-o(1)}$	$3/2 - \delta$ -approx	Sparse, unweighted	[5]
Radius	$n^{3-o(1)}$	Exact	Dense, weighted	[3]
Eccentricities	$n^{2-o(1)}$	$5/3 - \delta$	Sparse, unweighted	[5]
Reach Centrality	$n^{2-o(1)}$	$2 - \delta$	Sparse, unweighted	[3]
Reach Centrality	$n^{2-o(1)}$	Any finite	Sparse, weighted	[3]
Betweenness Centrality	$n^{2-o(1)}$	Any finite	Sparse, weighted	[3]
Betweenness Centrality	$n^{3-o(1)}$	Exact	Dense, weighted	[3]

Table 2: Known hardness results for the measures of Table 1. The hardness results are based on different conjectures. We refer to the discussion above for the specific details.

In this paper we will follow this active line of research on hardness in \mathbf{P} , basing hardness on the two following popular conjectures.

Conjecture 1 (Orthogonal Vectors Conjecture). *Let $A, B \subseteq \{0, 1\}^d$ be two sets of n boolean vectors of dimension $d = \omega(\log n)$. Then there exists no algorithm that can determine whether there is an orthogonal pair $a \in A, b \in B$ in time $O(n^{2-\varepsilon})$ for any $\varepsilon > 0$.*

The OV conjecture is implied by SETH [28] and has been used in several papers as an intermediate step for showing SETH-hardness.

Conjecture 2 (Hitting Set Conjecture). *Let $A, B \subseteq \{0, 1\}^d$ be two sets of n boolean vectors of dimension $d = \omega(\log n)$. Then there exists no algorithm that can determine whether there is an $a \in A$ that is not orthogonal to any $b \in B$ in time $O(n^{2-\varepsilon})$ for any $\varepsilon > 0$.*

We say that a vector $a \in A$ that is not orthogonal to any $b \in B$ is a *hitting set*, as it “hits” each vector of B . The HS conjecture was introduced by Abboud et al. [5] as an analog to the OV conjecture. One reason for introducing this conjecture is that Carmosino et al. [12] give evidence that problems such as radius and median cannot be shown to have SETH-hardness due to the nature of the quantifiers in this problem.

1.2 Hardness for graphs with constant max degree

An important direction for future research on fine-grained complexity of polynomial time problems, as noted by Abboud et al. [5] and Abboud and Dahlgaard in [2] is to understand the complexity of fundamental problems in restricted, but realistic classes of graphs. Such results for graphs of bounded treewidth was shown by Abboud et al. [5] and for planar graphs by Abboud and

Dahlgaard [2]. The result of Abboud and Dahlgaard even holds for planar graphs with constant maximum degree. Following up on this work, we concentrate on showing hardness for problems in graphs with constant maximum degree (bounded degree graphs). This family of graphs has received attention in several communities and in some cases exhibits better algorithms and data structures than graphs with unbounded degrees. This includes approximation of NP hard problems [22, 9, 21], labeling schemes [7, 8], property testing [17] and graph spanners [15]. Extending the results of conditional lower bounds from [3, 5, 26] to bounded degree graphs thus seems like a natural step in developing the theory of hardness in \mathbf{P} .

Challenges of reducing to constant degree graphs Many of the reductions mentioned above in Section 1.1 work for sparse graphs, and it is often assumed that such reductions translate directly to bounded degree graphs. In several papers (see e.g. [8]) sparse graphs are reduced to bounded degree graphs through vertex-splitting and insertion of weight zero edges. However, for the results of this paper we seek CLBs for *unweighted* bounded degree graphs, and thus introducing edges of weight 0 is not feasible. In [1] a near-linear lower bound is given for computing a $(3/2 - \delta)$ -approximation of diameter in the CONGEST model even in sparse networks. They also present a reduction to bounded-degree graphs, however this reduction does not preserve the approximation ratio of the lower bound and only excludes exact computation in bounded degree. This problem seems inherent in reductions from sparse graphs to bounded degree graphs: The most common reductions replace nodes with degree x by a balanced binary tree with x leaves, and each such leaf then has an edge to the binary trees of the corresponding neighbours. By reducing in this way two nodes in the new graph corresponding to the same node in the original graph may have $\Omega(\log n)$ distance, while two nodes corresponding to different original nodes may have distance 1. Such an imbalance can be addressed by introducing long paths in the graph, however this may introduce imbalance in approximation guarantees for problems like diameter, since the longest distance in the graph may now end “in the middle” of such a path. In the reductions in this paper we have to be extra careful to make sure that these issues do not occur.

1.3 Our results

In this paper we present tight hardness results for all of the measures of Table 1 in graphs with constant maximum degree. Our hardness results for betweenness centrality also improves on the result of [3] in general graphs by working also for unweighted graphs. More precisely we show the following theorems.

Theorem 1. *No algorithm can compute a $(3/2 - \delta)$ -approximation of the diameter of an unweighted constant degree graph in time $O(n^{2-\epsilon})$ for any $\epsilon, \delta > 0$ unless Conjecture 1 is false.*

Theorem 2. *No algorithm can compute a $(3/2 - \delta)$ -approximation of the radius of an unweighted constant degree graph in time $O(n^{2-\epsilon})$ for any $\epsilon, \delta > 0$ unless Conjecture 2 is false.*

Theorem 3. *No algorithm can compute a $(5/3 - \delta)$ -approximation of all eccentricities in an unweighted constant degree graph in time $O(n^{2-\epsilon})$ for any $\epsilon, \delta > 0$ unless Conjecture 1 is false.*

Theorem 4. *No algorithm can compute a $(2 - \delta)$ -approximation of the reach centrality of any node in an unweighted constant degree graph in time $O(n^{2-\epsilon})$ for any $\epsilon, \delta > 0$ unless Conjecture 1 is false.*

Theorem 5. *No algorithm can compute the betweenness centrality of any node in an unweighted constant degree graph in time $O(n^{2-\epsilon})$ for any $\epsilon > 0$ unless Conjecture 1 is false.*

All of our results above match the corresponding hardness results for sparse graphs, extending them to graphs with constant maximum degree. As mentioned all of these problems can be solved in $\tilde{O}(n^2)$ or even $O(n^2)$ time in unweighted graphs with constant max degree², thus all of our results are tight. Furthermore, a more efficient $\tilde{O}(m\sqrt{n})$ algorithm exists for 3/2-approximating the diameter of a graph [26, 13] and for 5/3-approximating all eccentricities of the graph [13]. Thus, our approximation guarantees are also optimal for these kinds of problems.

2 Hardness of eccentricity-related problems

In this section we will prove Theorems 1, 2, and 3. Our reductions are inspired by standard techniques as used in [26, 5, 14, 11] and many others. These reductions all use the *OV-graph* described below as a basis and add different shortcuts and nodes based on the problem at hand. In the case of constant degree graphs we have to be careful with the way we define deroutes, as we seek to get the same approximation guarantees as in the sparse case (see Table 2). A natural approach is to simply take the graph from eg. [26] and replace each node by a binary tree and each edge by a sufficiently long path (say length x). In [26] they create a graph where the diameter is either 3 or 2, and by doing this reduction we get a graph, where we would hope that the diameter is either $(3 + o(1))x$ or $(2 + o(1))x$. However, since the edges are replaced by paths we instead end up with either $(3 + o(1))x$ or $(\frac{5}{2} + o(1))x$ giving a much worse bound on the approximation. Below we show how to circumvent this bound by extending the *OV-graph* more carefully.

We will now define a general graph construction, the *OV-graph*, for reductions based on Conjecture 1, which has been used in several papers previously [4, 5, 14, 11, 3, 28]:

Given an instance $A, B \subseteq \{0, 1\}^d$ of the OV problem, we construct the *OV-graph* as follows:

- For each vector $a \in A$ and $b \in B$ create a node. We will overload the notation and use a and b to refer to both the vector and node when this is clear from the context.
- Create a set C of d nodes denoted c_1, c_2, \dots, c_d .
- For each node a create an edge (a, c_i) if $a[i] = 1$ for each $1 \leq i \leq d$. Do the same for each node b .

The following observation is key in all reductions using this graph.

Observation 1. *A pair $a \in A, b \in B$ is orthogonal if and only if $d(a, b) > 2$.*

For reductions based on Conjecture 2 we will also use the *OV-graph*, but since we are looking for an $a \in A$ that is *not* orthogonal to *any* $b \in B$, we will instead try to determine whether $d(a, b) = 2$ for all $b \in B$.

We note, that we will assume that each node a and b has an edge to at least on c_i . Otherwise this would be the all zeroes vector, and we can easily answer the OV problem.

2.1 Diameter in constant degree graphs

We are now ready to prove Theorem 1. Our reduction to diameter in constant degree graphs uses the *OV-graph* with the following modifications:

- For each node $u \in A \cup B$ insert a balanced binary tree with d leaves rooted at u . Denote the trees *vector trees* and the root vertices a_r and b_r respectively.

²If we assume unique or even polynomial number of shortest paths for BC.

- For each node $c \in C$ insert two balanced binary trees with n leaves, both rooted at c . One for handling edges to A , and one for B . Denote the tree for handling edges to A and B respectively the c_A -tree and the c_B -tree.
- For each edge (a, c) of the OV -graph, insert a path of length p (to be fixed later) from a leaf in the tree rooted at a_r , to a leaf in the c_A -tree. Do the same for edges (b, c) . We assign paths to leaves in such a way that each leaf has degree at most 2 in the final graph.
- Create a balanced binary tree with n leaves and connect each leaf to a node a_r with a path of length p , such that each a_r is connected to a distinct leaf. Do the same for the nodes b_r . Call these trees the A -shortcut tree and B -shortcut tree respectively.
- At each root a_r and b_r of a vector tree, add a path of length p and denote the end of each path by a_p .

We call this the OV_{dia} -graph. Figure 1 contains an example OV_{dia} -graph for reference. It is easy to verify that this graph has maximum degree 4 with only the nodes a_r , b_r , and c_i having this degree. We make the following claims about distances in the OV_{dia} -graph, the proofs can be

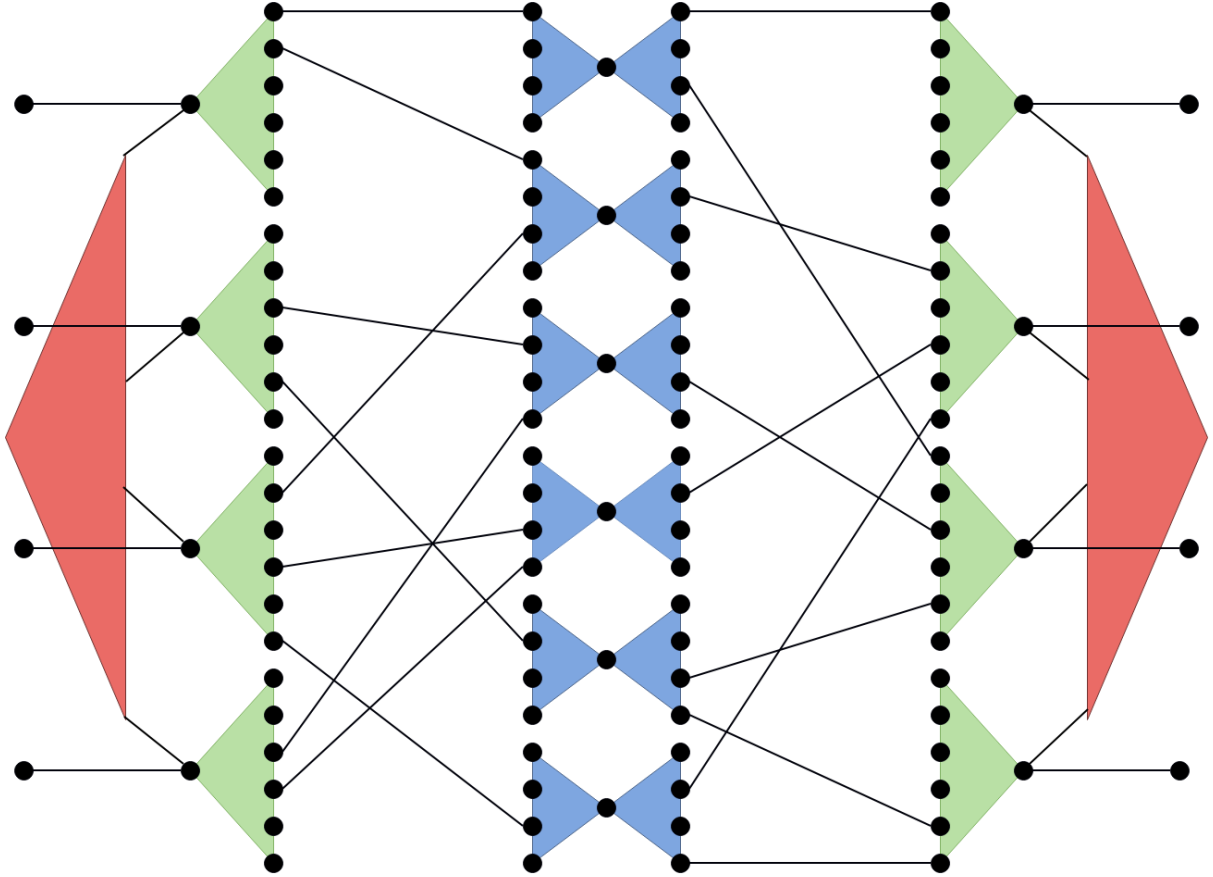


Figure 1: The OV_{dia} -graph, note that all lines in the graph are paths of length p . Green triangles are vector trees, blue triangles are c_A and c_B trees and red triangles are the A and B -shortcut trees

found in appendix A.1:

Claim 1. For two vectors $a \in A$, $b \in B$ that are not orthogonal, $d(a_p, b_p) \leq 4p + 2 \lg n + 2 \lg d$.

Claim 2. For two vectors $a \in A$, $b \in B$ that are orthogonal, $d(a_p, b_p) \geq 6p$.

Claim 3. For two nodes u and v in vector trees of A , $d(u, v) \leq 2p + 2 \lg n$. Symmetrically for B .

Claim 4. For a node u in some c_A or c_B tree, $\epsilon(u) \leq 4p + 4 \lg n + 2 \lg c$.

Claim 5. For a node u in the A -shortcut tree, $\epsilon(u) \leq 4p + 4 \lg n + 2 \lg c$.

The following lemma now follows directly from Claims 1 through 5 above and by picking $p = \omega(\log n)$ sufficiently large.

Lemma 1. The OV_{dia} -graph has diameter $(6 + o(1))p$ if an orthogonal pair exists, and $(4 + o(1))p$ if no orthogonal pair exists.

Using this Lemma we can now prove Theorem 1: Assume that there exists an algorithm with running time $O(n^{2-\epsilon})$ for computing a $(3/2 - \delta)$ -approximation of the diameter of a constant degree graph for some $\epsilon, \delta > 0$. We then create the OV_{dia} -graph for an instance of the OV problem, and run our algorithm. If no orthogonal pair exists the algorithm will return at most $(6 - \delta' + o(1))p \ll 6p$ for some fixed $\delta' > 0$ and we can thus correctly decide whether an orthogonal vector pair exists. Since the graph has $\tilde{O}(n)$ nodes and edges and can be constructed in $\tilde{O}(n)$ time, this implies a $O(n^{2-\epsilon})$ algorithm for the OV problem contradicting Conjecture 1.

2.2 Radius in constant degree graphs

In order to prove Theorem 2 we will use a similar approach as in the previous section, however we will be reducing from the HS problem of Conjecture 2. We will use the following graph, OV_{rad} :

Take two copies of the OV_{dia} -graph defined in the previous section, G_1 and G_2 and glue the graphs together in the nodes a_p , such that G_2 is a “mirrored” G_1 along the nodes a_p . This is illustrated in figure 2 below. Note that the minimum eccentricity must be in a node a_p since

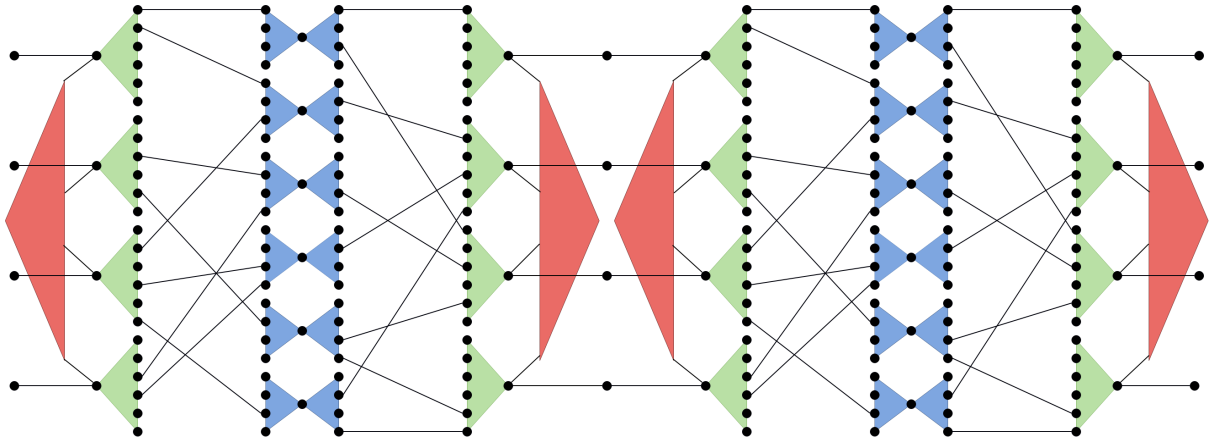


Figure 2: The OV_{rad} -graph, as in figure 1 all lines are paths of length p .

all paths from a node $u \in G_1$ to $v \in G_2$ must pass through a node a_p .

We now set $p = \omega(\log n)$ sufficiently large and get the following lemma:

Lemma 2. A node a_p in OV_{rad} has $\epsilon(a_p) = (4 + o(1))p$ if a is a hitting set and $\epsilon(a_p) = (6 + o(1))p$ if a is not a hitting set.

Proof. Since G_2 is a copy of G_1 it is enough to examine the eccentricities of a_p in G_1 . Assume that a forms a hitting set (that is a is not orthogonal to any $b \in B$). We then have from Claim 1 that $\epsilon(a_p) = 4p + O(\lg n)$.

Assume that a does not form a hitting set, since there then exists a $b \in B$ which is orthogonal to a , we have from Claim 2 that $\epsilon(a_p) \geq 6p$. \square

Now Theorem 2 follows from the above lemma in the same way that Theorem 1 followed from the discussion in the previous section.

2.3 Computing all eccentricities

For Theorem 3 we consider the graph from the proof of Theorem 1. From the discussion in Section 2.1 we see that some node a_r must have $\epsilon(a_r) = (5 + o(1))p$ if there exists an orthogonal vector pair and that all nodes a_r have $\epsilon(a_r) = (3 + o(1))p$ if no such pair exists. Thus, it follows that if an algorithm can compute a $(5/3 - \delta)$ approximation of all eccentricities in $O(n^{2-\epsilon})$ time for any $\epsilon, \delta > 0$ we have a contradiction to Conjecture 1.

3 Hardness of centrality problems

In this section we present Theorems 4 and 5.

3.1 Reach centrality

For reach centrality we will employ a slightly modified version of the diameter graph from Section 2.1. The goal will be to distinguish between a RC of $(3 + o(1))p$ and $(\frac{3}{2} + o(1))p$, thus ruling out a $(2 - \delta)$ -approximation for any $\delta > 0$.

Consider the OV_{dia} graph constructed in the proof of Theorem 1. We create a slightly modified graph, where the roots of the A and B -shortcut trees are connected by a path of length $2(p - \log n)$. We denote the middle node of this path by u . We will now consider the lengths of shortest paths passing through u . This is illustrated below in Figure 3

Claim 6. *Assume that no two vectors $a \in A$ and $b \in B$, from which the modified OV_{dia} graph was created, are orthogonal. Then $RC(u) = (\frac{3}{2} + o(1))p$.*

Proof. First note, that the distance from u to any node in one of the shortcut tree is at most p , thus for $RC(u)$ to be $(\frac{3}{2} + \delta)p$ there must exist nodes v, w with shortest path going through u , which are at distance at least $(\frac{1}{2} + \delta)p$ from the shortcut trees. Consider some node v on the path from some a_r to the A -shortcut tree at distance $(\frac{1}{2} - \delta)p$ from a_r and some node w on the path from some b_r to the B -shortcut tree at distance $(\frac{1}{2} - \delta)p$ from b_r . Since we assumed that no two vectors of the OV problem are orthogonal, it follows that $d(v, w) = (3 - 2\delta)p + o(p)$ by going through some node c_i , and the shortest path from v to w thus cannot pass through u . It is easy to see that any candidate shortest path for $RC(u)$ has to pass through such a pair of nodes v, w the claim follows. \square

If we now assume that there exists a pair of vectors $a \in A$ and $b \in B$ that are orthogonal, we know that $d(a_p, b_p) = (6 + o(1))p$. We will consider going from a_p to b_p and show that $RC(u) = (3 + o(1))p$.

Claim 7. *Assume that there exists $a \in A$ and $b \in B$ that are orthogonal. Then $RC(u) = (3 + o(1))p$.*

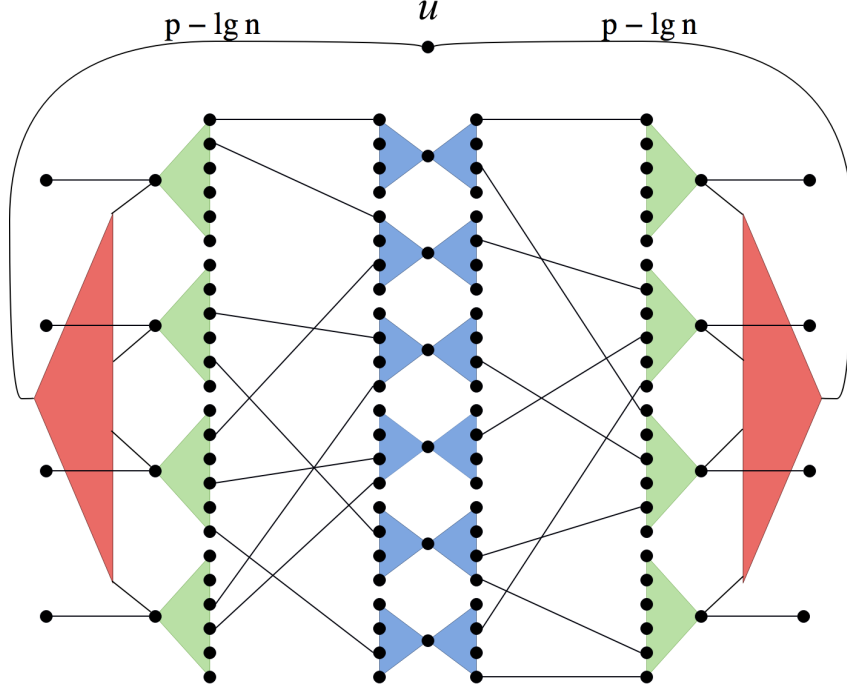


Figure 3: The modified OV_{dia} -graph for reach centrality

Proof. We consider the path from a_p to b_p going through u . This path has length $2(p + p + \log n + p - \log n) = 6p$. Observe also that any other path between a_p and b_p has length at least $6p$ by the proof of Theorem 1, thus this is a shortest path from a_p to b_p passing through u , and it follows that $RC(u)$ is at least $3p$. It is not hard to see that this is also an upper bound on $RC(u)$. \square

It now follows that any algorithm that can compute a $(2 - \delta)$ -approximation of $RC(u)$ in bounded degree graphs can distinguish between the two cases. Thus any algorithm that can do this in time $O(n^{2-\varepsilon})$ for any $\varepsilon > 0$ is a contradiction to Conjecture 1.

3.2 Betweenness centrality

We will start by showing Theorem 5 for sparse graphs instead of constant degree graphs, and then change the construction to work also in this case. We will base our reduction on the OV -graph and modify it as follows:

- Add two nodes x and y , and an edge (x, y) .
- Add an edge (a, x) for each $a \in A$, and edge (b, y) for each $b \in B$
- Create two copies G_1 and G_2 of this graph. Denote node $x \in G_1$ by x_1 and $x \in G_2$ by x_2 .
- Delete all nodes from B in G_1

Having created the graphs G_1 and G_2 we now query the betweenness centrality of x_1 in G_1 and of x_2 in G_2 . By comparing the two values we will be able to determine the answer to the OV problem instance.

Lemma 3. $BC(x_2) > BC(x_1)$ if and only if there is an orthogonal pair $a \in A$, $b \in B$.

Proof. If $BC(x_2) > BC(x_1)$ it must be the case that some shortest path P with a node $b \in B$ as an endpoint goes through x_2 , since these are the only paths not counted in $BC(x_1)$. The other endpoint of P cannot be y or any node $c \in C$ as these are at distance 1 from b . P must therefore have an endpoint in $a \in A$ and since P passes through x it must be of length 3. It now follows that a and b form an orthogonal vector pair. If no such pair a and b exists, all such paths P have length 2 and do not pass through x_2 . \square

We are now ready to prove the full version of Theorem 5. As for general unweighted graphs, our reduction for betweenness centrality in constant degree graphs, starts with the OV -graph, and modifies it as follows:

- As in the OV_{dia} -graph, add vector trees at each node $v \in A \cup B$ and tree-pairs c_A, c_B for each $c \in C$. Connect these exactly as in the OV_{dia} -graph.
- Insert a node x and make it the root of a balanced binary tree with n leaves. Connect each leaf of this x -tree to a root node a_r in a vector-tree $a \in A$ with a path of length p .
- Insert a node y and a y -tree connected to nodes in B exactly as the x -tree is with A .
- Insert a path of length p between x and y .
- Denote this graph by G_1 and create a copy G_2 . Denote node $x \in G_1$ by x_1 and $x \in G_2$ by x_2 .
- In G_2 create a node b' for each $b \in B$ and add the edge (b_r, b') for each such b' .

This construction is illustrated in Figure 4.

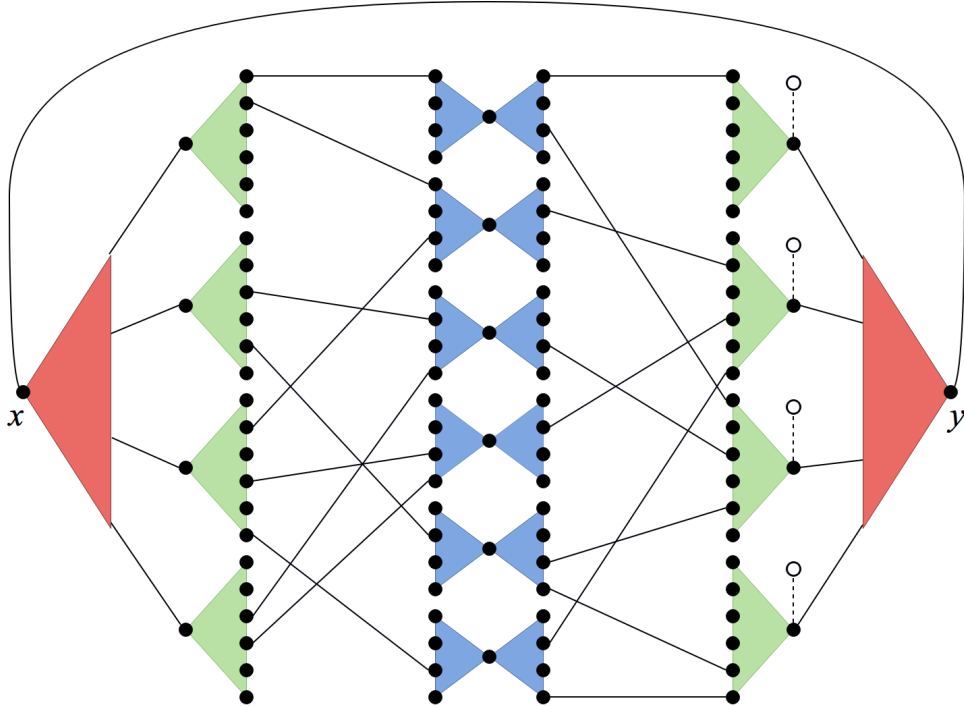


Figure 4: The graph for betweenness centrality in constant degree graphs. The white nodes along with dotted edges are the b' nodes, only in G_2 . All other lines are paths of length p .

We first examine some distances in G_1 which will be useful.

Observation 2. $d(a_r, b_r) = 2p + 2 \lg n + 2 \lg d$ when a and b are not orthogonal

Observation 3. $d(a_r, b_r) = 3p + 2 \lg n$ when a and b are orthogonal.

Using these observations we can prove the following lemma.

Lemma 4. $BC(x_2) > BC(x_1) + n^2 \cdot (\frac{1}{2}p + \lg d) + n \cdot (2n - 2)$ if and only if there is an orthogonal pair $a \in A, b \in B$.

Proof. We will examine the contribution to $BC(x_2)$ of each inserted node b' .

Consider a vector $b \in B$ and a vector $a \in A$ that are not orthogonal. The shortest path from b' to a_r has length $2p + 2 \lg n + 2 \lg d$ and does not go through x_2 . However on the path between a_r and x_2 there are a number of nodes v for which the shortest path between b' and v goes through x_2 . Since $d(b', x_2) = 2p + \lg n$ and $d(x_2, a_r) = p + \lg n$ we get that there are $\frac{p}{2} + \lg n + \lg d$ such nodes v .

If a vector $b \in B$ is not orthogonal to any $a \in A$ the node b' will then contribute $\frac{p}{2} + \lg n + \lg d$ to $BC(x_2)$ for each a for a total of $n(\frac{p}{2} + \lg n + \lg d)$ for each such vector b . Thus if there is no orthogonal pair the total contribution to $BC(x_2)$ from nodes b' will be $n^2(\frac{p}{2} + \lg n + \lg d)$. However, by doing this we have overcounted the contribution of b' and each node of the x -tree. Thus the actual contribution becomes $n^2 \cdot (\frac{p}{2} + \lg d) + n \cdot (2n - 2)$ as the tree has $2n - 2$ nodes if we exclude the root x_2 .

If a vector $b \in B$ is orthogonal to an $a \in A$ the shortest path from b' to a_r will go through x_2 and likewise for all nodes v on the path from x_2 to a_r for a total contribution of at least $p + \lg n$ to $BC(x_2)$

Observe that w.l.o.g. all the shortest paths considered above and thus each contribute 1 to the BC of x_2 . \square

Theorem 5 now follows directly from the above lemma using the same approach as in Section 2.1.

References

- [1] Amir Abboud, Keren Censor-Hillel, and Seri Khoury. Near-linear lower bounds for distributed distance computations, even in sparse networks. In *Proc. 30th International Symposium on Distributed Computing (DISC)*, pages 29–42, 2016.
- [2] Amir Abboud and Søren Dahlgaard. Popular conjectures as a barrier for dynamic planar graph algorithms. *CoRR*, abs/1605.03797, 2016. To appear at FOCS'16.
- [3] Amir Abboud, Fabrizio Grandoni, and Virginia Vassilevska Williams. Subcubic equivalences between graph centrality problems, APSP and diameter. In *Proc. 26th ACM/SIAM Symposium on Discrete Algorithms (SODA)*, pages 1681–1697, 2015.
- [4] Amir Abboud and Virginia Vassilevska Williams. Popular conjectures imply strong lower bounds for dynamic problems. In *Proc. 55th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 434–443, 2014.
- [5] Amir Abboud, Virginia Vassilevska Williams, and Joshua R. Wang. Approximation and fixed parameter subquadratic algorithms for radius and diameter in sparse graphs. In *Proc. 27th ACM/SIAM Symposium on Discrete Algorithms (SODA)*, pages 377–391, 2016.

- [6] Amir Abboud, Virginia Vassilevska Williams, and Huacheng Yu. Matching triangles and basing hardness on an extremely popular conjecture. In *Proc. 47th ACM Symposium on Theory of Computing (STOC)*, pages 41–50, 2015.
- [7] Noga Alon and Rajko Nenadov. Optimal induced universal graphs for bounded-degree graphs. *CoRR*, abs/1607.03234, 2016.
- [8] Stephen Alstrup, Søren Dahlgaard, Mathias Bæk Tejs Knudsen, and Ely Porat. Sublinear distance labeling. In *Proc. 24th European Symposium on Algorithms (ESA)*, pages 5:1–5:15, 2016.
- [9] Per Austrin, Subhash Khot, and Muli Safra. Inapproximability of vertex cover and independent set in bounded degree graphs. *Theory of Computing*, 7(1):27–43, 2011.
- [10] Ulrik Brandes. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25:163–177, 2001.
- [11] Massimo Cairo, Roberto Grossi, and Romeo Rizzi. New bounds for approximating extremal distances in undirected graphs. In *Proc. 27th ACM/SIAM Symposium on Discrete Algorithms (SODA)*, pages 363–376, 2016.
- [12] Marco L. Carmosino, Jiawei Gao, Russell Impagliazzo, Ivan Mihajlin, Ramamohan Paturi, and Stefan Schneider. Nondeterministic extensions of the strong exponential time hypothesis and consequences for non-reducibility. In *Proc. 7th Innovations in Theoretical Computer Science*, pages 261–270, 2016.
- [13] Shiri Chechik, Daniel H. Larkin, Liam Roditty, Grant Schoenebeck, Robert Endre Tarjan, and Virginia Vassilevska Williams. Better approximation algorithms for the graph diameter. In *Proc. 25th ACM/SIAM Symposium on Discrete Algorithms (SODA)*, pages 1041–1052, 2014.
- [14] Søren Dahlgaard. On the hardness of partially dynamic graph problems and connections to diameter. In *Proc. 43rd International Colloquium on Automata, Languages and Programming (ICALP)*, pages 48:1–48:14, 2016.
- [15] Fedor V. Fomin, Petr A. Golovach, and Erik Jan van Leeuwen. Spanners of bounded degree graphs. *Inf. Process. Lett.*, 111(3):142–144, 2011.
- [16] Linton C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977.
- [17] Oded Goldreich and Dana Ron. Property testing in bounded degree graphs. *Algorithmica*, 32(2):302–343, 2002. See also STOC’97.
- [18] Ronald J. Gutman. Reach-based routing: A new approach to shortest path algorithms optimized for road networks. In *Proc. 6th Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 100–111, 2004.
- [19] Per Hage and Frank Harary. Eccentricity and centrality in networks. *Social networks*, 17(1):57–63, 1995.
- [20] S. L. Hakimi. Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations Research*, 12(3):450–459, 1964.

- [21] Magnús M. Halldórsson. Approximations of weighted independent set and hereditary subset problems. *J. Graph Algorithms Appl.*, 4(1), 2000.
- [22] Eran Halperin. Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs. *SIAM Journal on Computing*, 31(5):1608–1623, 2002. See also SODA’00.
- [23] Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-sat. *Journal of Computer and System Sciences*, 62(2):367–375, 2001.
- [24] M. E. J. Newman. A measure of betweenness centrality based on random walks. *Social Networks*, 27(1):39–54, 2005.
- [25] Mihai Patrascu. Towards polynomial lower bounds for dynamic problems. In *Proc. 42nd ACM Symposium on Theory of Computing (STOC)*, pages 603–610, 2010.
- [26] Liam Roditty and Virginia Vassilevska Williams. Fast approximation algorithms for the diameter and radius of sparse graphs. In *Proc. 45th ACM Symposium on Theory of Computing (STOC)*, pages 515–524, 2013.
- [27] Gert Sabidussi. The centrality index of a graph. *Psychometrika*, 31(4):581–603, 1966.
- [28] Ryan Williams. A new algorithm for optimal 2-constraint satisfaction and its implications. *Theoretical Computer Science*, 348(2-3):357–365, 2005.

A Omitted proofs

A.1 proofs of claims about the OV_{dia} graph

Proof of claim 1

Proof. Any path from a_p to b_p must go through both of the length p paths starting at the two nodes. Since a and b are not orthogonal, there exists a node c such that there is an path of length p from the vector tree at a to c_A and likewise from b ’s vector tree to c_B , taking this path adds the height of the vector trees, the height of c_A and c_B and $2p$ to the total path, giving $d(a_p, b_p) \leq 4p + 2 \lg n + 2 \lg d$ \square

Proof of claim 2

Proof. The distance between a_p and b_p must be greater when the two corresponding vectors are orthogonal, as they do not have direct paths to any node c . The shortest path between the two must then go through some other vector trees, w.l.o.g. assume that a' and b' are *not* orthogonal, and a' is the closest such vector tree to a_r , either in the shortcut tree or through some c_A tree. The distance between a_r and a'_r through the shortcut tree is at least $2p$, and thus adds $2p$ to the distance from claim 1. To see that this is the shortest distance, note that this is the shortest possible path using the shortcut trees to get from a to a' , and any path going through a clause would use $> 2p$ edges to get to a' . \square

Proof of claim 3

Proof. For any two nodes u in the vector tree of a vector a and v in the vector tree of a' the path through the a_r , a'_r and the shortcut tree is at most $2p + 2 \lg d + 2 \lg n$. \square

Proof of claim 4

Proof. We consider two cases: either a node a_p is the furthest from u (symmetric for b_p), or a node v in some c'_A or c'_B is the furthest from u .

For the first case any such node can be reached by traversing trees c_A and c_B , a path of length p to a vector tree rooted at a'_r (possibly $a = a'$ in which case the path is shorter.), a "shortcut" of $2p + 2 \lg n$ edges through the shortcut tree to a_r , and p edges from a_r to a_p . For a total of $3p + 3 \lg n + \lg d$

For the second case consider the following path from u to v , let a be a vector for which there is a length p path from a 's vector tree to c_A , similarly let a' be such a vector for c'_A , the path from u , to v through a_r the shortcut tree and a'_r must then have length at most $4p + 6 \lg n + 2 \lg c$ since $\lg n$ is both the height of the shortcut tree and the trees c_A and c_B . \square